

Attorney Docket No: IDF 1538 (4000-01800)

**REMARKS**

Applicants have reviewed the Office Action dated November 14, 2003. Applicants respectfully submit that the art provided in the Office Action does not address the claims of Applicants. In brief summary, the art cited in the Office Action (U.S. Patent No. 6,269,373 hereinafter "Apte") provides no disclosure or teaching of how to set or modify transactional behavior for a CORBA method or of a method for propagating the transactional context in the service context of an IIOP message. The only significant mentions of transactional behavior occur in column 7, lines 43-49, which indicates that Enterprise Java Beans delegate the ability to provide transactional behavior to the container in which they are placed but does not discuss the setting of the transactional behavior for the method, and in column 15, lines 29-31, which indicates that an EJB server provides management of distributed transactions as a service. These are Java specific references discussing, in a very general way, how to provide transactional services, but not at all teaching the claimed method of setting or modifying whether transactional services will be used for a given CORBA method.

Selected specific citations in the Office Action will be briefly addressed in the following discussion.

**In paragraph 4 of the Office Action:**

Apte in Column 7 is cited as disclosing a method for setting transactional behavior, but the cited references only mentions that EJB's may have transactional behavior with no discussion of how to set or change it.

Apte in Column 10 is cited as disclosing "a client creating a transaction policy." There are two difficulties with this citation; first it appears to disclose a client obtaining an object reference, either by creating the reference, looking up the reference in a naming service, or obtaining the reference from a string. There is no mention of creating a transaction policy defining transactional behavior of CORBA methods. Additionally, the focus in the cite is on the client, but the claim language actually refers to "a system *remote from a client* creating a transaction policy" not the client creating the policy.

Apte in Column 7 is cited for the premise that a deployment descriptor is translated to create a transaction policy. Apte in Column 7 only discloses that Enterprise Java Beans may

Attorney Docket No: IDP 1538 (4000-01800)

have a deployment descriptor about the bean to be read by a tool. The deployment descriptor file as claimed is discussed and defined in the paragraph ending page 22 and continuing to page 23, among other places in the application. The claimed deployment descriptor file is not specific to EJB's, and while it shares the nomenclature applied in the cited reference, it clearly has a different definition as provided in the specification of the present application. Further, there is no disclosure in this section of the reference that a deployment descriptor is used to create a transaction policy which may effect one or a group of CORBA methods, instead the deployment descriptor in the cited reference is apparently used to address unnamed descriptions about a specific Enterprise Java Bean. This, if anything, teaches away from the claimed language.

Apte in Column 8 is cited for the premise that an interceptor remote from the client checks the transaction policy with respect to a method name. There is no mention of transactions and no mention of checking a transaction policy in this section of Apte. The section does discuss the ORB seeking a match to the request of the client, but if anything this teaches away from the current invention, as the client request does not define whether the method should be transactional, but it is instead set in the deployment descriptor file remote from the client.

Apte in Column 12 is cited for the promise that the decision to complete a control object interpositioning is defined by the results of a check of the transaction policy with respect to the method name being invoked. The cited section describes the client looking up an appropriate name to call to accomplish an intended purpose. First, it does not mention a transaction policy check whatsoever or even discuss transactional behavior. Even if it did, the suggestion in the reference is that the client would look up a method that includes or does not include transactional behavior, thus defining the behavior on the client side. This teaches directly away from the claimed invention that it is not the client invocation that sets behavior for the CORBA method but the transaction policy created from the deployment descriptor file remote from the client.

Paragraph 5 of the Office Action provides no new basis for rejection.

In Paragraph 6 & 7 of the Office Action:

Apte in Column 7 is cited for creating a transaction policy on a system remote from the client during deployment of that system or in response to an IIOP message. The cited section makes no mentions of transactions or transaction policies, and appears to simply discuss the

Attorney Docket No: IDI: 1538 (4000-01800)

deployment of java beans without discussion of when or how a transaction policy might be implemented at deployment or in run-time to define when the deployment descriptor file is checked.

In Paragraph 8 of the Office Action:

Apte in Column 7 is cited as disclosing a method for changing transactional behavior, but the cited references only mentions that EJB's may have transactional behavior with no discussion of how to set or change it.

Apte in Column 17 lines 30-33 is cited as disclosing defining transactional behavior for a CORBA method, but the cited section makes no mention of transactional behavior or how to define it.

Apte in Column 7 is cited for the premise that a deployment descriptor is translated to create a transaction policy. Apte in Column 7 only discloses that Enterprise Java Beans may have a deployment descriptor about the bean to be read by a tool. The deployment descriptor file as claimed is discussed and defined in the paragraph ending page 22 and continuing to page 23, among other places in the application. The claimed deployment descriptor file is not specific to EJB's, and while it shares the nomenclature applied in the cited reference, it clearly has a different definition as provided in the specification of the present application. Further, there is no disclosure in this section of the reference that a deployment descriptor is used to create a transaction policy which may effect one or a group of CORBA methods, instead the deployment descriptor in the cited reference is apparently used to address unnamed descriptions about a specific Enterprise Java Bean. This, if anything, teaches away from the claimed language.

Apte in Column 8 is cited for the premise that invocations of the CORBA method results in a defined transactional behavior based on the transaction policy. There is no mention of transactions and no mention of checking a transaction policy in this section of Apte. The section does discuss the ORB seeking a match to the request of the client, but if anything this teaches away from the current invention as the client request does not define whether the method should be transactional, but it is instead set in the deployment descriptor file remote from the client as reflected in the transaction policy created from the deployment descriptor file.

Apte in Column 7 is cited as disclosing modifying the deployment descriptor file to

Attorney Docket No: IDF 1538 (4000-01800)

change the transactional behavior for the CORBA method. The cited reference discusses the adoption of platform independent java beans into platform specific EJB's. It does not discuss transactions or transactional behavior at all and certainly not the modification of transactional behavior by the modification of a deployment descriptor file as defined and claimed in the present application.

Apte in Column 18 combined with Apte in Column 12 and Examiner's inherency observation are cited as disclosing identical invocations from identical client objects resulting in different transactional behavior based on a modified policy. Applicant respectfully notes that the inherency observation made in the Office Action discusses that similar objects may have different transactional behavior, but this is precisely the point of this claim, that it is identical objects (not merely similar) that yield different transactional results before and a change in the transaction policy. Apte in Column 18 is not addressing transactional behavior but only the process of stringification and destringification and the importance of the processes working the same across platforms. Apte in Column 12, describes the client typecasting the returned generic object to the necessary class type. First, it does not mention a transaction policy check whatsoever or even discuss transactional behavior. Even if it did, the suggestion in the reference is that the client would typecast a method that includes or does not include transactional behavior, thus defining the behavior on the client side. This teaches directly away from the claimed invention that it is not the client invocation that sets behavior for the CORBA method but the transaction policy created from the deployment descriptor file remote from the client.

In Paragraph 9 of the Office Action:

Apte in Column 7 with Apte in Column 10 is cited as disclosing a deployment descriptor that defines transactional behavior for more than one CORBA method. But Apte in Column 7 only discloses that Enterprise Java Beans may have a deployment descriptor about the bean to be read by a tool. The deployment descriptor file as claimed is discussed and defined in the paragraph ending page 22 and continuing to page 23, among other places in the application. The claimed deployment descriptor file is not specific to EJB's, and while it shares the nomenclature applied in the cited reference, it clearly has a different definition as provided in the specification of the present application. Further, there is no disclosure in this section of the reference that a deployment descriptor is used to create a transaction policy which will effect a group of CORBA

Attorney Docket No: IDF 1538 (4000-01800)

methods as claimed in this claim, instead the deployment descriptor in the cited reference is apparently used to address unnamed descriptions about a specific Enterprise Java Bean. This, if anything, teaches away from the claimed language.

In Paragraph 10, the exact same issues arise as just discussed with respect to Paragraph 9; only now the claim is for all methods on the server, not just more than one method on the server.

Paragraphs 11, 12, & 13 all address claims that define different forms or locations of the deployment descriptor file and the transaction policy. The cited sections of Apte discuss how to obtain an object reference (Column 10) and how to manage persistence of a bean (Column 16 and Column 15), not how to set or modify transactional behavior of a CORBA method.

In Paragraph 14, the exact same issues arise as just discussed with respect to Paragraph 9; only now the claim is for creating policies for multiple servers.

In Paragraph 15 of the Office Action:

Apte in Column 1 is cited as teaching a method for propagating transactional context, but Apte in Column 1 only discusses that ORBS may provide communication. There is no discussion of transactions at all or of the specific challenge of how to propagate a transactional context.

Apte in Column 8 is cited as disclosing an interceptor on a system local to the client inserting an object representing the transaction context on the service context of the IIOP message and for disclosing an interceptor remote from the client extracting the object representing the transaction context from the service context of the IIOP message. Apte in Column 8 only discusses finding an object reference and provides no discussion of transactions or transactional context and certainly no specific teaching of how to propagate a transactional context by inserting it onto the service context of an IIOP message and later extracting it from the service context of an IIOP message using interceptors.

Conclusion:

In summary, the cited reference simply does not address modifying and setting transactional behavior for CORBA methods, or propagation of transactional context. To the extent it could even logically be extended to opine on such topics, it leads one to anticipate a

Attorney Docket No: IDF 1538 (4000-01800)

client driven process teaching away from the related claims of the present application which focus on a process for setting and modifying transactional behavior independent of the client view of the client invocation.

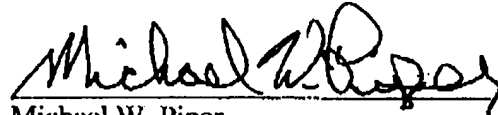
Applicants respectfully submit that the present application is in condition for allowance. If the Examiner has any questions or comments or otherwise feels it would be helpful in expediting the application, he is encouraged to telephone the undersigned at (972) 731-2288.

The Commissioner is hereby authorized to charge payment of any further fees associated with any of the foregoing papers submitted herewith, or to credit any overpayment thereof, to Deposit Account No. 21-0765, Sprint.

Date: 2/17/2004

CONLEY ROSE, P.C.  
5700 Granite Parkway, Suite 330  
Plano, Texas 75024  
(972) 731-2288  
(972) 731-2289 (facsimile)

Respectfully submitted,



Michael W. Piper  
Reg. No. 39,800

ATTORNEY FOR APPLICANT